Network Working Group M. Crispin
Request for Comments 748 SU-AI
NIC 44125 1 April 1978

TELNET RANDOMLY-LOSE Option

1. Command name and code.

RANDOMLY-LOSE 256

2. Command meanings.

IAC WILL RANDOMLY-LOSE

The sender of this command REQUESTS permission to, or confirms that it will, randomly lose.

IAC WON'T RANDOMLY-LOSE

The sender of this command REFUSES to randomly lose.

IAC DO RANDOMLY-LOSE

The sender of this command REQUESTS that the receiver, or grants the receiver permission to, randomly lose.

IAC DON'T RANDOMLY-LOSE

The command sender DEMANDS that the receiver not randomly lose.

3. Default.

WON'T RANDOMLY-LOSE

DON'T RANDOMLY-LOSE

i.e., random lossage will not happen.

4. Motivation for the option.

Several hosts appear to provide random lossage, such as system crashes, lost data, incorrectly functioning programs, etc., as part of their services. These services are often undocumented and are in general quite confusing to the novice user. A general means is needed to allow the user to disable these features.

5. Description of the option.

The normal mode does not allow random lossage; therefore the system is not allowed to crash, mung user files, etc. If the server wants to provide random lossage, it must first ask for permission from the user by sending IAC WILL RANDOMLY-LOSE.

If the user wants to permit the server to randomly lose, it replys with IAC DO RANDOMLY-LOSE. Otherwise it sends IAC DONT RANDOMLY-LOSE, and the server is forbidden from randomly losing.

Alternatively, the user could request the server to randomly lose, by sending IAC DO RANDOMLY-LOSE, and the server will either reply with IAC WILL RANDOMLY-LOSE, meaning that it will then proceed to do some random lossage (garbaging disk files is recommended for an initial implementation). Or, it could send IAC WONT RANDOMLY-LOSE, meaning that it insists upon being reliable.

Since this is implemented as a TELNET option, it is expected that servers which do not implement this option will not randomly lose; ie, they will provide 100% reliable uptime.